WHAT IS CLAIMED IS:

1. A protocol stored on a computer-readable medium for transmitting commands and data between a bus controller and a network device interface across a common digital network, wherein said protocol comprises a set of low level instructions for sending respective commands and data such that it is possible to implement at least one of the bus controller and the network device interface as a state machine.

2. A protocol according to Claim 1, wherein said protocol comprises a set of low level instructions such that one of the instructions cause at least one of the bus controller and network device interface to perform a single operation only.

3. A protocol according to Claim 1, wherein said protocol uses low level instructions requiring less processing such that at least one of the NDI device and the bus controller is a state machine implemented as either an Application Specific Integrated Circuit (ASIC) or a field programmable gate array (FPGA).

4. A protocol according to Claim 1, wherein said protocol includes at least one of a command and a data structure for sending respective commands and arguments to the network device.

5. A protocol according to Claim 1, wherein said protocol uses messages having bit lengths that vary based on at least one of the command and data being transmitted in the message.

6. A protocol according to Claim 1, wherein said protocol uses messages that are independent of a information representing a block size for the message or a checksum for the message.

7. A protocol according to Claim 1, wherein said protocol uses messages that contain data representing a sync pattern that is used to synchronize with the message.

8. A protocol according to Claim 1, wherein said protocol uses messages that contain data representing a flag bit that indicates if an error condition exists in the network device interface.

10. A protocol according to Claim 1, wherein said protocol uses messages that contain data representing at least one parity bit that is used to check for errors in the message.

11. A protocol according to Claim 1, wherein if a command includes more than one instruction, said protocol separates the command into each separate instruction and sends each instruction one at a time.

12. A protocol according to Claim 1, wherein said protocol is transmitted in one of a Manchester encoded format and a Universal Asynchronous Receiver Transmitter (UART) format protocol.

13. A protocol according to Claim 1, wherein said protocol sends commands in the form of a command frame comprising:

bits representing a command sync pattern;

bits representing an encoded address;

bits representing an encoded command; and

an encoded parity bit.

14. A protocol according to Claim 1, wherein said protocol sends argument information in an argument frame comprising:

bits representing a data sync pattern;

bits representing an encoded argument;

an encoded flag bit; and

an encoded parity bit.

15. A protocol according to Claim 1, wherein said protocol sends data information in a data frame comprising:

bits representing a data sync pattern;

bits representing encoded data;

an encoded flag bit; and

an encoded parity bit.

16. A method for transmitting commands and data between a bus controller and a network device interface across a common digital network comprising the step of transmitting commands and data using a set of low level instructions such that it is

possible to implement at least one of the bus controller and the network device interface as a state machine.

17. A method according to Claim 16, wherein said transmitting step comprises transmitting commands and data using a set of low level instructions such that one instruction causes at least one of the bus controller and network device interface to perform a single operation only.

18. A method according to Claim 16, wherein said transmitting step uses protocol that includes at least one of a command and a data structure for sending respective commands and arguments to the network device.

19. A method according to Claim 16, wherein said transmitting step uses protocol having messages with bit lengths that vary based on at least one of the command and data being transmitted in the message.

20. A method according to Claim 16, wherein said transmitting step uses protocol that contains messages that are independent of a information representing a block size for the message or a checksum for the message.

21. A method according to Claim 16, wherein said transmitting step uses protocol containing messages that contain data representing a sync pattern that is used to synchronize with the message.

22. A method according to Claim 16, wherein said transmitting step uses protocol containing messages that contain data representing a flag bit that indicates if an error condition exists in the network device interface.

23. A method according to Claim 16, wherein said transmitting step uses protocol containing messages that contain data representing at least one parity bit that is used to check for errors in the message.

24. A method according to Claim 16, wherein in said transmitting step if a command includes more than one instruction, said transmitting step separates the command into each separate instruction and sends each instruction one at a time.

25. A method according to Claim 16, wherein said transmitting step transmits commands and data in one of a Manchester encoded format and a Universal Asynchronous Receiver Transmitter (UART) format protocol.

26. A method according to Claim 16, wherein said transmitting step sends commands in the form of a command frame comprising:

bits representing a command sync pattern;

bits representing an encoded address;

bits representing an encoded command; and

an encoded parity bit.

27. A method according to Claim 16, wherein said transmitting step sends argument information in an argument frame comprising:

bits representing a data sync pattern;

bits representing an encoded argument;

an encoded flag bit; and

an encoded parity bit.

28. A method according to Claim 16, wherein said transmitting step sends data information in a data frame comprising:

bits representing a data sync pattern;

bits representing encoded data;

an encoded flag bit; and

an encoded parity bit.